Software Architectures for Morphing in Polymorphous Computing Architectures

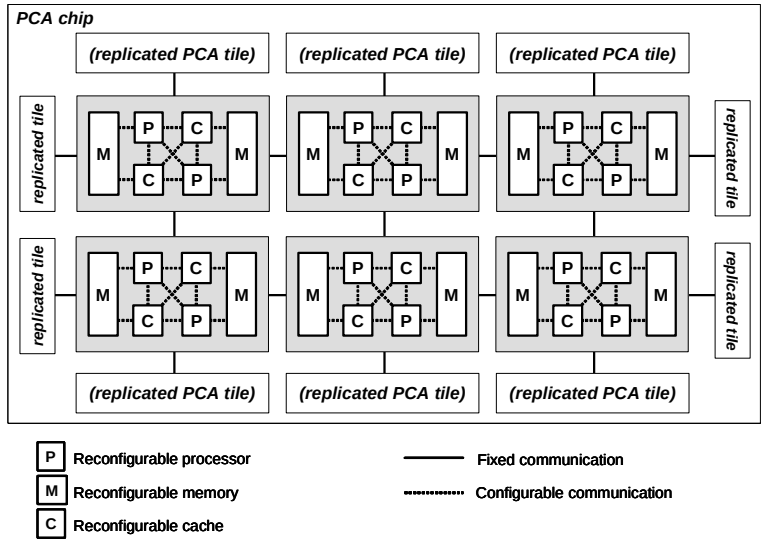
Dan Campbell, Mark Richards
Georgia Tech

Dennis Cottel, Randy Judd
USN SSC-SD

High Performance Embedded Computing Workshop
28 September 2004

Morphing in PCA Architectures

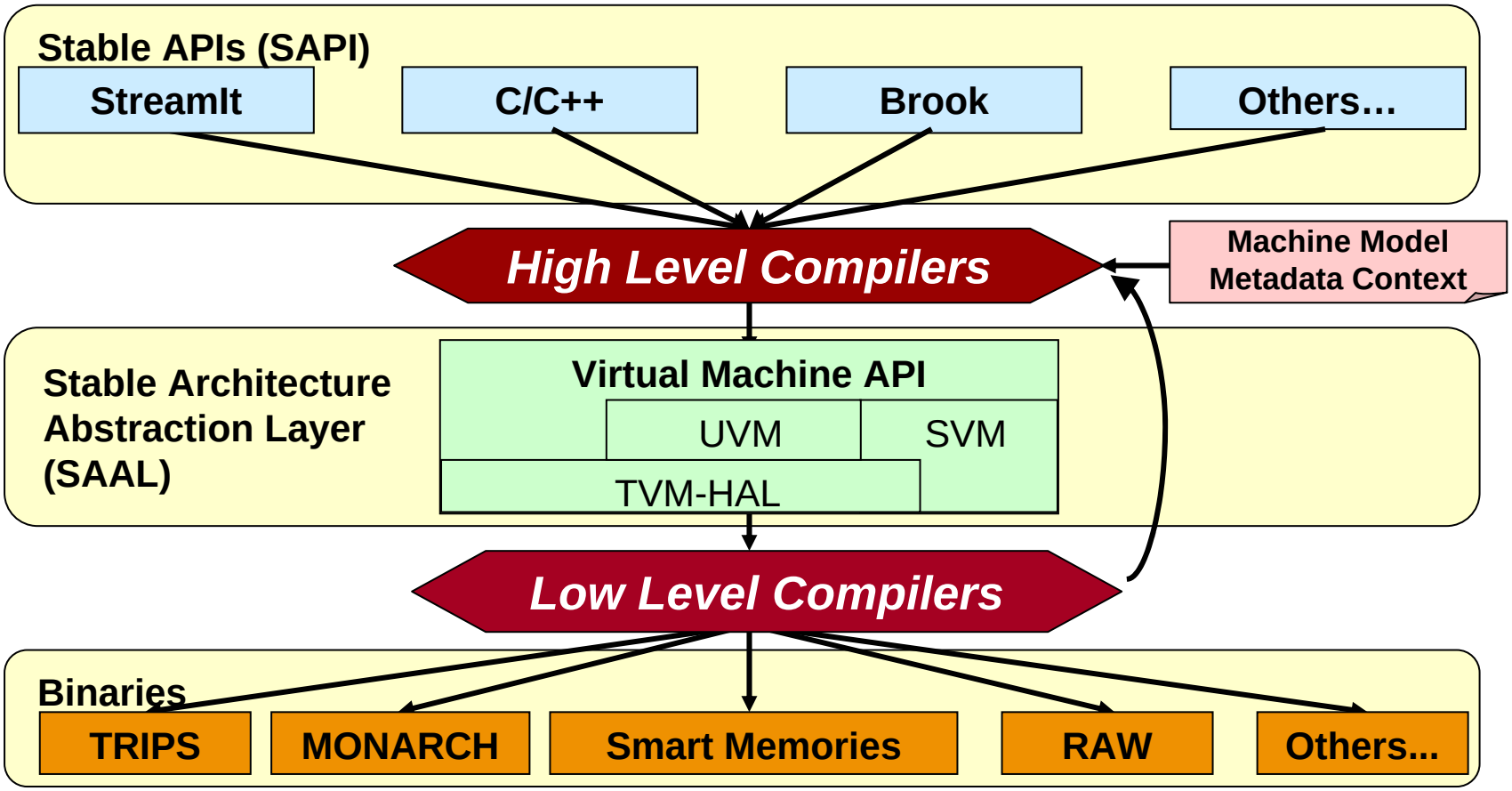
- **The DARPA Polymorphous Computing Architectures (PCA) program is developing embedded high-performance computing platforms with strong, rapid reconfigurability**
- **PCA processors are essentially "multiprocessors on a chip"**
  - **tiled architectures**
  - **reconfigurable processing aggregates**
  - **reconfigurable networks**



- **"Morphing" is the reconfiguration and re-allocation of PCA hardware resources within a chip in response to various events**
  - **key capability to achieve PCA goals**
  - **portability across PCA chips must be maintained**

PCA Two-Level Module Compilation Architecture

- **Two-level compile + customizable machine models enables targeting of same functionality to multiple machine configurations**



PCA Morph Taxonomy Dimensions

- A taxonomy of morph possibilities has been established by the Morphware Forum
- Morphs distinguished by
  - Initiator: application, run-time system, or compiler-generated code
  - Resource constancy: static, or changed
  - Component persistence: continues, or is replaced

	Run-time System		Application		Compiling System	
	Components continue	*Components change*	*Components continue*	*Components change*	*Components continue*	*Components change*
Resource allocation doesn't change	Type 0a Run-time environment changes the way it uses the running application.	Type 1a Run-time system changes components to different but equivalent resources.	Type 2a Application makes API call to give up or gain some resources.	Type 3a Application makes API call to change processing to do so within existing resource set.	Type 4a Compiler instructions reconfigure allocated resources.	Type 5a Compiler switches to a different library able to use the same resources.
Resource allocation changes	Type 0b Run-time system changes resource allocation of a running application transparently to the application.	Type 1b Run-time system configures resources and loads components at application startup.	Type 2b Application makes API call to give up or gain some resources.	Type 3b Application makes API call to add or replace one or more components using different resources.	Type 4b Compiler requests different resources to meet change in performance specified by metadata.	Type 5b Compiler switches to a different library that uses different resources.